UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/602,425 | 06/24/2003 | William G. Pagan | RPS920030088US1/2858P | 9632 |

47052 7590 11/14/2008

IBM RP-RPS
SAWYER LAW GROUP LLP
2465 E. Bayshore Road, Suite No. 406
PALO ALTO, CA 94303

| EXAMINER | |
|---|---|
| NUNEZ, JORDANY | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2175 | |

| NOTIFICATION DATE | DELIVERY MODE |
|---|---|
| 11/14/2008 | ELECTRONIC |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patent@sawyerlawgroup.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

Application Number: 10/602,425
Filing Date: June 24, 2003
Appellant(s): PAGAN, WILLIAM G.

Joseph A. Sawyer Jr.
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed 08/28/2008 appealing from the Office action mailed 05/15/2008.

**(1) Real Party in Interest**

A statement identifying by name the real party in interest is contained in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The statement of the status of claims contained in the brief is correct.

**(4) Status of Amendments After Final**

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

**(5) Summary of Claimed Subject Matter**

The summary of claimed subject matter contained in the brief is correct.

**(6) Grounds of Rejection to be Reviewed on Appeal**

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

**(7) Claims Appendix**

The copy of the appealed claims contained in the Appendix to the brief is correct.

**(8) Evidence Relied Upon**

| | | |
|---|---|---|
| 20030090471 | Slaunwhite et al. | 05-2003 |
| 5999895 | Forest | 12-1999 |

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-4, 6, 12-16, 23-27, 35, 37, 39, 41, 43, 45, 49-52 are rejected under 35 U.S.C. 102(e) as being anticipated by Slaunwhite et al. (US20030090471, hereinafter Slaunwhite).

As to claims 1, 12, Slaunwhite shows:

A method comprising steps, and a corresponding computer-readable storage medium storing program instructions, for providing a hot key corresponding to a particular function (e.g., zoom) in a computer system, the computer system having a graphical user interface (GUI) and including a pointing

device enabling a user to select items displayed in the GUI, the particular function provided for a context of an application program, a user providing input within a context (abstract, lines 1-7), comprising:

integrating a hot key configuring function into the GUI such that a user can access the hot key configuring function from within the context and without leaving the context (page 3, paragraph [0040], lines 5-12) (e.g., user can quickly press hot key, change settings, and then go back to application, without looking for a toolbar or a large dialog), wherein the context includes a displayed item (e.g., a zoom GUI) displayed in the GUI corresponding to the particular function, and wherein the particular function is performed in response to the displayed item being selected by the pointing device (page 3, paragraph [0050]) (e.g., figures 4 and 5 show that the zoom GUI item appears in two places, in the top toolbar and as element 202; one of ordinary skill in the art would readily understand that the zoom GUI item in the top toolbar would be displayed both before the appearance of element 202 in response to a user pressing a hot key, and after a user dismissal of element 202; the zoom GUI item in the top toolbar would enable a user to "perform the particular function when the item is selected by the pointing device");

mapping the hot key to the particular function and storing the mapping (e.g., assigning of the shortcut keys to item types), the mapping and storing performed without the user leaving the context and in response to the user utilizing the hot key configuring function in the context (page 2, paragraph [0037]; figure 3), wherein the mapping causes the particular function (e.g., zoom) to be accessed by the computer system when the mapped hot key is selected (page 3, paragraph [0038], last 4 lines), and wherein the mapping includes:

receiving an indication of the particular function (e.g., zoom) to which the hot key (e.g., short cut key) is to be mapped, the indication provided by the user moving the pointing device over the displayed item (e.g., receiver 102 receives the item type from the user input unit 10) to indicate the particular function (e.g., zoom) corresponding to the displayed item (e.g., non-command item type zoom GUI) for the mapping, wherein the same displayed item is selectable by the pointing device to perform the particular function and is indicatable by the pointing device to indicate the particular function for the mapping (page 2, paragraph [0036]; figure 3, element 150) (e.g., figures 4 and 5 show that the zoom GUI item appears in two places, in the top toolbar and as element 202; one of ordinary skill in the art would

readily understand that the zoom GUI item in the top toolbar would be both selectable to perform the zooming function, and indicatable by a pointing device, or user input unit 10, to indicate the zooming function for mapping, per paragraph [0036], lines 1-3); and

receiving a key combination as the hot key in response to the user selecting the key combination using a hardware input device, the key combination being received after the indication of the particular function to which the hot key is to be mapped has been received (e.g., the key receiver receives the shortcut key after receiving the non-command item type from user input 10) (page 2, paragraph [0036]).

As to claims 2, 13, 24, Slaunwhite shows:

The method of claim 1 further comprising the step of:

accounting for ambiguities in the receiving of the key combination, such that the mapping of the particular function to the hot key is not confused with accessing a function previously mapped to the hot key (page 3, paragraph [0038]) (e.g., when the user presses a shortcurt key, it is unambiguously decided which function is being accessed, in accordance with the previous shortcut key assignation step) .

As to claims 3, 14, 25, Slaunwhite shows:

wherein the pointing device includes a mouse (page 2, paragraph [0027]), and wherein the hot key configuring function integrating step further includes the steps of:

determining the plurality of items selectable in the context (page 2, paragraphs [0035] and [0036], lines 1-3);

and providing (e.g., making possible but not necessarily causing) a mechanism that maps at least one of the plurality of items to the hot key from the context without the user leaving the context (page 2, paragraph [0036], lines 3-8).

As to claims 4, 15, 26, Slaunwhite shows:

wherein the hot key configuring function integrating step further includes a mechanism that accounts for ambiguities, if any, between the hot key (e.g., command item) and a pre-existing hot key (e.g., non-command item) (page 3, paragraph [0038]).

As to claims 6, 16, 27, Slaunwhite shows:

wherein the indication of the particular function using the pointing device over the displayed item does not cause the particular function to be performed (page 2, paragraph [0036]) (e.g., a user identifying the non-command item type to the item receiver 102 does not trigger the function).

As to claim 23, Slaunwhite shows:

A computer system (page 4, paragraph [0058]) comprising:

a hardware mechanism that provides an application, the application providing a context (figure 4) and having a particular function (e.g., zoom) available therein, the particular function provided for a context of an application program (e.g., zooming), a user providing input within the context (e.g., arrow down)(page 3, paragraph [0050], lines 1-12);

a graphical user interface (GUI) (figure 4);

and a hot key configuring function integrated into the GUI (figure 1, element 100) such that a user can access the hot key configuring function from within the context and without leaving the context (page 3, paragraph [0040], lines 5-12) (e.g., user can quickly press hot key, change settings, and then go back to application, without looking for a toolbar or a large dialog), the integrated hot key configuring function utilized by a user to designate a map of the hot key to the particular function and store the mapping without the user leaving the context (e.g., assigning of the shortcut keys to item types), wherein the mapping causes the particular function to be accessed by the computer system when the mapped hot key is selected (page 3, paragraph [0038], last 4 lines),, and wherein the mapping is created by receiving an indication of the particular function to which the hot key is to be mapped and receiving a key combination as the hot key in response to the user selecting the key combination using a hardware input device, the key combination being received after the indication of the particular function to which the hot key is to be

mapped has been received (page 2, paragraph [0036), wherein the indication is provided by the user moving the pointing device over the displayed item to indicate the particular function corresponding to the displayed item for the mapping (page 2, paragraph [0036), wherein the same displayed item is selectable by the pointing device to perform the particular function and is indicatable by the pointing device to indicate the particular function for the mapping (e.g., figures 4 and 5 show that the zoom GUI item appears in two places, in the top toolbar and as element 202; one of ordinary skill in the art would readily understand that the zoom GUI item in the top toolbar would be both selectable to perform the zooming function, and indicatable by a pointing device, or user input unit 10, to indicate the zooming function for mapping, per paragraph [0036], lines 1-3).

As to claims 35, 39, 43, Slaunwhite shows:

The method of claim 1 wherein mapping the hot key to the particular function without the user leaving the context includes receiving the indication of the particular function (e.g., item type) made by the user without the user providing input to a menu separate from the context (column 2, paragraph [0036]; figure 3, elements 150) (e.g., no menu is taught).

As to claims 37, 41, 45, Slaunwhite shows:

wherein the displayed item is a text-based item including displayed text (e.g., the text indicating zoom level), and wherein the indication of the particular function includes selecting a portion of the text of the corresponding item, the portion of the text being less than the entire displayed text of the display item (e.g., setting focus to the zoom level selects a portion of the text of the item indicating the zoom level) (page 3, paragraph [0040], lines 1-8; page 1, paragraph [0005], lines 1-9]).

As to claim 49, Slaunwhite shows:

The method of claim 35 wherein the context is a particular context, and wherein the application program has a plurality of different contexts which can each independently receive user input (page 1, paragraph [0004]).

As to claims 50, 51, 52, Slaunwhite shows:

wherein the indication of the particular function to which the hot key is to be mapped is provided by an action of the pointing device different than an action of the pointing device providing the selection of the displayed item to perform the particular function (page 2, paragraph [0036) (e.g., one of ordinary skill in the art would readily understand that an action of user input unit 10 would be different when selecting the zoom GUI item for zooming action than when indicating the zoom GUI item for mapping, per paragraph [0036], lines 1-3).

Claims 46, 47, 48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Slaunwhite et al. (US20030090471, hereinafter Slaunwhite).

As to claims 46, 47, 48, Slaunwhite shows:

Slaunwhite shows a method, computer-readable storage and system substantially as claimed, as specified above.

Slaunwhite further teaches: a shortcut key being Alt-Z and having been assigned the "zoom drop down listbox" item type (page 3, paragraph [0050]) (Thus, Slaunwhite teaches the letter Z as part of the shortcut key to the Zoom function).

Slaunwhite fails to specifically show: wherein the indicating of the particular function for the mapping includes clicking on one letter of the text of the corresponding displayed item with the pointing device, wherein a key of the hardware input device that matches the one letter of the text is assigned as a portion of the hot key.

It would have been obvious to one of ordinary skill in the art, having the teachings of Slaunwhite at the time that the invention was made, to have included the indicating of the particular function for the

mapping includes clicking on one letter of the text of the corresponding displayed item with the pointing device, wherein a key of the hardware input device that matches the one letter of the text is assigned as a portion of the hot key with the method, computer-readable storage and system as taught by Slaunwhite.

One would have been motivated to make such combination because a way to simplify the way in which a user accesses a particular non-command user interface item would have been obtained and desired, as expressly taught by Slaunwhite (page 1, paragraph [0010]).

Claims 10, 21, 32, 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Slaunwhite et al. (US20030090471, hereinafter Slaunwhite) in view of Forest (US5999895).

As to claims 10, 21, 32, 36:

Slaunwhite shows a method, computer-readable storage and system substantially as claimed, as specified above.

Slaunwhite further shows: wherein the computer system further includes a pointing device (page 2, paragraph [0027]), wherein the context includes a displayed feature corresponding to the particular function (page 2, paragraph [0037]) (e.g., item type on a list) and wherein the mapping step further includes the steps of:

receiving an indication of the particular function to which the hot key is to be mapped (page 2, paragraph [0037]) (inherent to "user selects the item type from a list")

receiving a selection of a key combination as the hot key (page 2, paragraph [0036], lines 3-8).

Slaunwhite fails to specifically show: moving the pointing device over the displayed item to indicate the particular function includes the user hovering the pointing device over the displayed item for a predetermined amount of time, wherein the indication of the particular function for mapping does not include clicking the pointing device on the displayed item, and wherein the selection of the displayed item to perform the particular function includes clicking on the displayed item; and indication of the particular function includes the user hovering a pointing device over a portion of the corresponding item in the GUI for a predetermined amount of time, and wherein the indication of the particular function for mapping does

not include clicking the pointing device on the displayed item, and wherein the selection of the displayed item to perform the particular function includes clicking on the displayed item

Forest shows that it was well-known, at the time of the instant invention, that squares of a keyboard may be sized to match an operator's abilities, and also that an on-screen keyboard may be used, with a picture of a keyboard drawn on a computer display; the operator then selecting a letter by pointing to that letter's key image on the display with a mouse, then indicating that he has reached his target either by operating a switch (e.g., clicking) or by maintaining the location indicated by the pointed (e.g., dwelling or hovering) on the key image for a predetermined period of time (column 3, lines 53-66) (e.g., Forest teaches indicating by hovering, or not clicking, and selecting by clicking were both well known in the art prior to the instant invention).

Thus, one of ordinary skill in the art could have combined the elements as claimed by known methods (e.g., as taught by Slaunwhite and Forest), and in combination, each element merely would have performed the same function as it did separately (see KSR, 550 U.S. at ___, 82 USPQ2d at 1391).

References to specific columns, figures or lines should not be limiting in any way. The entire reference provides disclosure related to the claimed invention.

References to specific columns, figures or lines should not be limiting in any way. The entire reference provides disclosure related to the claimed invention.

(**10) Response to Argument**

The Office action Dated 5/15/2008 contained the following typographical errors:

1) The statement of page 2 stating:

Claims 1-4, 6, 12-16, 23-27, 34, 35, 37-39, 41-43, 45-49 are rejected under 35 U.S.C. 102(e) as being anticipated by Slaunwhite et al. (US20030090471, hereinafter Slaunwhite).

Should have instead read:

Claims 1-4, 6, 12-16, 23-27, 35, 37, 39, 41, 43, 45, 49-52 are rejected under 35 U.S.C. 102(e) as being anticipated by Slaunwhite et al. (US20030090471, hereinafter Slaunwhite).

2) Claims 34, 38, 42, which were rejected on bottom of page 5 and top of page 6, should have not been addressed because they were previously canceled.

3) The statement of page 7 stating:

"Claims 36, 40, 44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Slaunwhite et al. (US20030090471, hereinafter Slaunwhite).

As to claims 36, 40, 44, Slaunwhite shows:"

Should have instead read:

"Claims 46-48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Slaunwhite et al. (US20030090471, hereinafter Slaunwhite).

As to claims 46-48, Slaunwhite shows:"

4) The statement of page 8 stating:

"Claims 10, 21, 32, 36, 40, 44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Slaunwhite et al. (US20030090471, hereinafter Slaunwhite) in view of Forest (US5999895).

As to claims 10, 21, 32, 36: "

Should have instead read:

"Claims 10, 21, 32, 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over Slaunwhite et al. (US20030090471, hereinafter Slaunwhite) in view of Forest (US5999895).

As to claims 10, 21, 32, 46:"

Please note that the typographical errors mentioned above are all related to the numbering of claims in the Final Office Action of 5/15/2008. The 5/15/2008 Office Action properly addressed the claim limitations themselves. Appellant seemly acknowledges that these were typographical errors, and not a

change in the reasons for rejection, by having failed to point it out said errors. Further, this Examiner's Answer continues the same line of reasoning used in the 5/15/2008 Office Action, and thus does not introduce new grounds for rejection.

Applicant's arguments have been fully considered but are not persuasive. Examiner reiterates that references to specific columns, figures or lines should not be limiting in any way. The entire reference provides disclosure related to the claimed invention. Appellan argues that:

The Slaunwhite Reference And Claims 1-4, 12-15, 23-26, 35, 39, 43, 49.

1) Claim 1 is patentable over Slaunwhite since the features of claim 1 are not disclosed or suggested by that reference. In particular, Slaunwhite fails to disclose or suggest the recited feature of a particular function being performed when a corresponding displayed item is selected by the pointing device, and the particular function of the displayed item is indicated for mapping when the pointing device is moved over that displayed item (page 9, antepenultimate paragraph).

Slaunwhite discloses that, when assigning a hot key to a function of the computer system, an item receiver 102 receives the item type from the user input unit 10 and receives an identification of a shortcut key from the user input unit 10 (paragraph [0036]). The user input unit 10 typically includes a keyboard and a mouse (paragraph [0027]). Slaunwhite teaches only one way for the item receiver 102 to receive the item type from the user input unit 10, in paragraph [0037]: a customization dialog where the user selects the item type from a list of available item types and then keys in the shortcut key that is associated with it. Thus, Slaunwhite teaches the standard way of mapping hot keys, where a separate list or menu of functions is displayed (i.e., the customization dialog) and the user selects a function in the list to which to assign a hot key (page 9, last paragraph).

Examiner disagrees.

Slaunwhite (page 2, paragraphs [0030] and [0033]) clearly teaches a particular function (e.g., a task) being performed when a corresponding displayed item s selected (i.e., selecting a user interface item, which is an instance of an item type, from a toolbar) by the pointing device (Note: Appellant does

not dispute this). Moreover, Slaunwhite (page 2, paragraph [0035]) teaches that there could be several instances of the same item type displayed simultaneously. Further, Slaunwhite teaches (page 3, paragraph [0037]) that a user typically (e.g., not always) selects an item type from a customization dialog containing a list of available item types and then keys in the shortcut key that is associated with it. Then, as one of ordinary skill in the art would understand, different item instances of the same item type are displayed both in the toolbar and the customization dialog. Thus, Slaunwhite clearly teaches a particular function (e.g., task) being performed when a corresponding displayed item is selected by the pointing device (e.g., performing a task by selecting an item type on a toolbar), and the particular function (e.g., task associated with the item type) of the displayed item being indicated for mapping when the pointing device is moved over that displayed item (e.g., selecting the particular function for mapping a shortcut when the item type is selected from a list on a customization dialog).

Moreover, Slaunwhite must be read for all it teaches. In particular, Slaunwhite teaches (page 2, paragraph [0036]) that when defining a shortcut key assignment, an item receiver (e.g., program module) receives the item type from a user input unit (e.g., by selecting item type with a mouse), without mentioning any customization dialog. As stated above, Slaunwhite (page 2, paragraphs [0030] [0035]) teaches that an item type instance may be displayed on a toolbar and that an item type instance may be selected using a mouse. In other words, Slaunwhite teaches that an item receiver may receive the item type by selecting an item type instance from a toolbar with a mouse (e.g., user input unit). Thus, one of ordinary skill in the art would readily understand that Slaunwhite teaches at least two ways of selecting the particular function for mapping a shortcut: a) typically by selecting item type instance a list on a customization dialog and b) when an item type instance is selected from a toolbar.

2) Slaunwhite is silent to Applicant's claimed feature described above. For example, the items displayed in Slaunwhite's mapping customization dialog are nowhere taught to be selectable to perform a corresponding function; they are provided as a list of items, each item describing a function which can be assigned to a keyed-in shortcut key if the item is selected. The displayed windows and boxes of Slaunwhite such as the zoom drop down listbox 200 perform functions, but these same displayed items

do not provide a dual ability allowing a user to both perform the corresponding function by selecting that item as well as indicate functions for mapping to a received hot key using a pointing device over the item, as claimed by Applicant (page 10, penultimate paragraph).

Examiner disagrees.

Appellant's statement "the items displayed in Slaunwhite's mapping customization dialog are nowhere taught to be selectable to perform a corresponding function" is incorrect, at least because it contradicts another of Appellant's statement "Slaunwhite teaches only one way for the item receiver 102 to receive the item type from the user input unit 10, in paragraph [0037] a customization dialog where the user selects the item type from a list of available item types and then keys in the shortcut key that is associated with it." In other words, the items displayed in Slaunwhite's customization dialog are "selectable to perform a corresponding function" of being received by the item receiver in order to define a shortcut key assignment.

3) Slaunwhite' s only example of indicating a particular function for mapping by using a pointing device is in paragraph [0037], which discloses selecting an item type in the list displayed in the customization dialog. The items in this list are nowhere disclosed to be also selectable to perform a function. The standard method to allow hot key mapping is to select a function from a list of functions and input a hot key to assign to that selected function, as indicated in Applicant's specification page 2, and this is what Slaunwhite teaches. The function types listed in such a list do not perform their function when selected--these items are presented for the sole purpose of assigning their function to a hot key, and not as a way to perform the function they represent. Nowhere does Slaunwhite disclose or suggest a displayed item that can be both selected with a pointing device to perform a corresponding function, and that same displayed item also indicated with the pointing device to indicate the corresponding function for mapping to a received hot key (page 11, last paragraph).

Examiner disagrees.

As explained above, Examiner points out that the limitation of claim 1 "selectable to perform a function" can be reasonably interpreted as "causing receipt by an item receiver of an item type for

defining a shortcut." Thus, Slaunwhite (pages 2-3, paragraphs [0036], [0037] clearly teaches the items in customization dialog list are indeed disclosed to be also selectable to perform a function (e.g., selecting an item type form the customization dialog causes an item receiver to receive the item type in order to define a shortcut key).

4) Paragraph [0036] and Fig. 3 element 150 only disclose that the item receiver 102 generally receives the item type from the user input unit 10. This "receiving" of an item type is only described in any detail in Slaunwhite as the selection of an item type from the customization dialog of paragraph [0037]. Nowhere does Slaunwhite disclose that the user is selecting a displayed function-performing item, such as a button on a tool bar, or a slider control, to provide that item type to the item receiver 102 for mapping to a hot key (page 11, penultimate paragraph).

Examiner disagrees.

As explained above, Slaunwhite must be read for all it teaches. In particular, Slaunwhite teaches (page 2, paragraph [0036]) that when defining a shortcut key assignment, an item receiver (e.g., program module) receives the item type from a user input unit (e.g., by selecting item type with a mouse), without mentioning any customization dialog. Further, Slaunwhite (page 2, paragraphs [0030] [0035]) teaches that an item type instance may be displayed on a toolbar and that an item type instance may be selected using a mouse. Combining the two teachings, one of ordinary skill in the art would readily understand that Slaunwhite teaches that an item receiver may receive the item type by selecting an item type instance from a toolbar with a mouse (e.g., user input unit). Thus, one of ordinary skill in the art would readily understand that Slaunwhite teaches at least two ways of selecting the particular function for mapping a shortcut: a) Appellant's acknowledged typical selection of an item type instance on a list of a customization dialog and b) selecting an item type instance from a toolbar.

5) The Examiner stated that Figs. 4 and 5 show a zoom GUI item that appears in a top toolbar and as element 202, and that one of ordinary skill would readily understand that the zoom GUI item in the top toolbar would be both selectable to perform the zooming function and indicatable by a pointing device

to indicate the zooming function for mapping. However, nowhere is the zoom GUI item described as allowing a user to both select the item to perform a function and indicate the item to associate its function for mapping to a received shortcut key when a pointing device is moved over it, as claimed by Applicant. One of ordinary skill would not understand that the item can perform both these features, since the standard way to map hotkeys is as disclosed by Slaunwhite: using a separate dialog with a list of functions. Slaunwhite provides no suggestion other than to use this standard dialog method; his descriptions about receiving an item identification are so broad as to suggest nothing beyond his disclosed dialog, and all of his displayed GUI items (except his mapping customization dialog) do not relate to performing mapping of hotkeys to functions). One of ordinary skill reading Slaunwhite would not have been led to the recited combination and would not have learned anything but a general, standard method of assigning the hot key to a function, and would have instead focused on the types of items being assigned to hot keys, such as the non-command type of user interface items that are the focus of Slaunwhite's disclosure (paragraph [0010]).

Examiner disagrees.

As explained above, Slaunwhite must be read for all it teaches. In particular, Slaunwhite teaches (page 2, paragraph [0036]) that when defining a shortcut key assignment, an item receiver (e.g., program module) receives the item type from a user input unit (e.g., by selecting item type with a mouse), without mentioning any customization dialog. Further, Slaunwhite (page 2, paragraphs [0030] [0035]) teaches that an item type instance may be displayed on a toolbar and that an item type instance may be selected using a mouse. Combining the two teachings, one of ordinary skill in the art would readily understand that Slaunwhite teaches that an item receiver may receive the item type by selecting an item type instance from a toolbar with a mouse (e.g., user input unit). Thus, one of ordinary skill in the art would readily understand that Slaunwhite teaches at least two ways of selecting the particular function for mapping a shortcut: a) Appellant's acknowledged typical selection of an item type instance on a list of a customization dialog and b) selecting an item type instance from a toolbar.

6) Claims 2 and 4 are further patentable over Slaunwhite since Slaunwhite does not disclose accounting for ambiguities in the receiving of the key combination such that the mapping of the particular function to the hot key is not confused with accessing a function previously mapped to the hot key; and claim 35 is further patentable since Slaunwhite does not disclose receiving the indication of the particular function without the user providing input to a menu separate from the context (page 12, last paragraph).

Examiner disagrees.

Appellant seemly merely concludes that Slaunwhite does not teach the limitations recited, without providing a reason as to why reaching that conclusion is reasonable. Examiner counters that Slaunwhite (page 3, paragraph [0038]) teaches that when the user presses a shortcut key, it is unambiguously decided as a function that is being accessed, in accordance with a previous shortcut key assignation step. Therefore shortcut accessing cannot be confused with shortcut key mapping, and vice versa. Further, as stated above, Slaunwhite teaches (page 2, paragraph [0036]) that when defining a shortcut key assignment, an item receiver (e.g., program module) receives the item type from a user input unit (e.g., by selecting item type with a mouse), without mentioning any customization dialog.

The Slaunwhite Reference And Claims 6, 16, and 2

7) Claim 6 is separately patentable over Slaunwhite. Claim 6 recites that the indication of the particular function using the pointing device over the displayed item does not cause the particular function to be performed. This shows that the indication of an item for mapping its function to a received hot key is not the same as selecting an item for performing its function. For example, a hovering action as disclosed in Applicant's specification (page 8, lines 7-12) can be used for indicating a function for mapping to a hot key, not performing the function. Slaunwhite does not disclose or suggest a displayed item that can be both selected with a pointing device to perform a corresponding function, and also indicated with the pointing device to indicate the corresponding function for hot key mapping, where the indication of the particular function does not cause the function to be performed. For example, Slaunwhite's zoom GUI item described above does not allow a user to indicate its function for shortcut key mapping when a

pointing device is moved over it, where such an indication does not cause the function of the zoom GUI window to be performed, as claimed by Applicant (page 14, first paragraph).

Examiner disagrees.

As stated above, Slaunwhite (page 2, paragraphs [0030] and [0033]) clearly teaches a particular function (e.g., a task) being performed when a corresponding displayed item s selected (i.e., selecting a user interface item, which is an instance of an item type, from a toolbar) by the pointing device (Note: Appellant does not dispute this). Moreover, Slaunwhite (page 2, paragraph [0035]) teaches that there could be several instances of the same item type displayed simultaneously. Further, Slaunwhite teaches (page 3, paragraph [0037]) that a user typically (e.g., not always) *selects (e.g., by moving pointing device over it)* an item type from a customization dialog containing a list of available item types and then keys in the shortcut key that is associated with it. Then, as one of ordinary skill in the art would understand, different item instances of the same item type are displayed both in the toolbar and the customization dialog. Thus, Slaunwhite clearly teaches a particular function (e.g., task) being performed when a corresponding displayed item is selected by the pointing device (e.g., performing a task by selecting an item type on a toolbar), and the particular function (e.g., task associated with the item type) of the displayed item being indicated for mapping when the pointing device is moved over that displayed item (e.g., selecting the particular function for mapping a shortcut when the item type is selected from a list on a customization dialog).

Moreover, Slaunwhite must be read for all it teaches. In particular, Slaunwhite teaches (page 2, paragraph [0036]) that when defining a shortcut key assignment, an item receiver (e.g., program module) receives the item type from a user input unit (e.g., by selecting item type with a mouse), without mentioning any customization dialog. As stated above, Slaunwhite (page 2, paragraphs [0030] [0035]) teaches that an item type instance may be displayed on a toolbar and that an item type instance may be selected using a mouse. In other words, Slaunwhite teaches that an item receiver may receive the item type by selecting an item type instance from a toolbar with a mouse (e.g., user input unit). Thus, one of ordinary skill in the art would readily understand that Slaunwhite teaches at least two ways of selecting

the particular function for mapping a shortcut: a) typically by selecting item type instance a list on a customization dialog and b) when an item type instance is selected from a toolbar.

The Slaunwhite Reference And Claims 37, 41, and 45

8) Claim 37 is separately patentable over Slaunwhite. Claim 37 recites that the displayed item is a text-based item including text, and the indication of the particular function includes selecting a portion of the text of the corresponding item, the portion of the text being less than the entire displayed text of the displayed item. These features are not disclosed or suggested by Slaunwhite. Slaunwhite does not disclose or suggest selecting a portion of text of a displayed item to indicate a function for mapping, where that displayed item is also selectable to perform a function. For example, the Examiner stated in the Final Office Action of 5/15/08 that Slaunwhite discloses this feature as setting focus to the zoom level selecting a portion of the text of the item indicating the zoom level (paragraphs [0040], [0005]). However, the displayed item indicating the zoom level is not disclosed or suggested as being used to indicate a function for hot key mapping. Slaunwhite mentions or suggests nothing about mapping a function indicated by selecting a portion of text (less than the entire text) of a displayed item with a pointing device, to a hot key (page 15, paragraph starting with "Claim 37 is [...]").

Examiner disagrees.

As explained above, Slaunwhite teaches (page 2, paragraph [0036]) that when defining a shortcut key assignment, an item receiver (e.g., program module) receives the item type from a user input unit (e.g., by selecting item type with a mouse), without mentioning any customization dialog. Further, Slaunwhite (page 2, paragraphs [0030] [0035]) teaches that an item type instance may be displayed on a toolbar and that an item type instance may be selected using a mouse. Combining the two teachings, one of ordinary skill in the art would readily understand that Slaunwhite teaches that an item receiver may receive the item type by selecting an item type instance from a toolbar with a mouse (e.g., user input unit). Thus, one of ordinary skill in the art would readily understand that Slaunwhite teaches at least two ways of selecting the particular function for mapping a shortcut: a) Appellant's acknowledged typical selection of an item type instance on a list of a customization dialog and b) selecting an item type

instance from a toolbar. Further, Slaunwhite teaches (as shown in figure 5) that when a user sets focus to a zoom level interface type, a portion of the text of the item indicating the zoom level is selected. Because Slaunwhite shows this being part of selecting a user interface item type, one of ordinary skill in the art would readily understand that when user sets focus to a zoom level interface type, thereby selecting a portion of the text of the item indicating the zoom level, the zoom level is indeed being used to indicate a function for shortcut key mapping.

The Slaunwhite Reference And Claims 50, 51, and 52

9) Claim 50 is separately patentable over Slaunwhite. Claim 50 recites that the indication of the particular function to which the hot key is to be mapped is provided by an action of the pointing device different than an action of the pointing device providing the selection of the particular function. For example, a hovering action over an item to indicate its function for mapping, as disclosed in Applicant's specification (page 8, lines 7-12), is one example of an action that is different than the standard clicking action performed on GUI display items such as buttons, menu items, etc. to perform the functions of those items. Slaunwhite does not disclose or suggest different actions of the pointing device on the same displayed item one action to perform the particular function, and the other action to indicate the particular function to be mapped to the hot key. For example, the Examiner stated that Slaunwhite discloses this feature at paragraph [0036], and that one of ordinary skill in the art would readily understand that an action of a user input unit 10 would be different when selecting the zoom GUI item for zooming action than when indicating the zoom GUI item for mapping. However, paragraph [0036] mentions no specifics regarding different actions for a particular displayed item, and Slaunwhite mentions or suggests nothing about allowing an action of a pointing device for the zoom GUI item (or any other displayed item) to map functions to hotkeys, the pointing device action being different than the function-performing pointing device action for that zoom-GUI item. Slaunwhite's disclosure focusses on providing keyboard focus on a displayed non-command item (e.g., paragraph [0051]), and is not concerned with different pointing device actions for a displayed item.

Examiner disagrees.

As stated above, Slaunwhite (page 2, paragraphs [0030] and [0033]) clearly teaches a particular function (e.g., a task) being performed when a corresponding displayed item s selected (i.e., selecting a user interface item, which is an instance of an item type, from a toolbar) by the pointing device (Note: Appellant does not dispute this). Further, Slaunwhite teaches (page 3, paragraph [0037]) that a user typically (e.g., not always) *selects (e.g., by moving pointing device over it)* an item type from a customization dialog containing a list of available item types and then keys in the shortcut key that is associated with it. Thus, Slaunwhite clearly teaches different pointing device actions for a displayed item (e.g., one when selecting from a toolbar, another when selecting from a list on a customization dialog).

The Slaunwhite Reference And Claims 46, 47, and 48

10) Claim 46 is separately patentable over Slaunwhite. Claim 46 recites that indicating the function for mapping includes clicking on one letter of the text of the corresponding displayed item with the pointing device, wherein a key of the hardware input device matching the one letter of the text is assigned as a portion of the hot key. These features are not disclosed or suggested by Slaunwhite. For example, Slaunwhite maps the shortcut key of "Alt-Z" to the function of displaying the zoom drop-down box shown in Fig. 4, but mentions or suggests nothing about mapping that display function to the shortcut key by selecting one letter of text on a displayed item that, when selected, also can perform the display function of the drop-down box (page 18, last paragraph).

Examiner disagrees.

As mentioned in the rejection of these claims, one would have been motivated to make such combination because a way to simplify the way in which a user accesses a particular non-command user interface item would have been obtained and desired, as expressly taught by Slaunwhite (page 1, paragraph [0010]). Examiner readily admits to Slaunwhite not teaching indicating of the particular function for the mapping by clicking on one letter of the text of the corresponding displayed item with the pointing device, wherein a key of the hardware input device that matches the one letter of the text is assigned as a portion of the hot key (e.g., see the rejection of claims in questions). However, absent evidence that the

modifications necessary to effect the combination of elements is uniquely challenging or difficult for one of ordinary skill in the art, the claim is unpatentable as obvious under 35 U.S.C. 103(a). Ex Parte Smith, 83 USPQ.2d at 1518-19 (BPAI, 2007) (citing KSR, 127 S.Ct. at 1740, 82 USPQ2d at1396.

The Slaunwhite Reference And Claims 10, 21, 32, and 36

11) Claim 10 is patentable over Slaunwhite since the feature of claim 10 is not disclosed or suggested by that reference. Claim 10 recites that indicating the particular function includes the user hovering the pointing device over the displayed item (or a portion thereof) for a predetermined amount of time, and the indication of the particular function for mapping does not include clicking the pointing device on the displayed item, and the selection of the displayed item to perform the particular function includes clicking on the displayed item. For example, the selection to perform the item's function can be a standard clicking action on a displayed item to perform a function, as is well known. Applicant' s recited use of the pointing device in selection of a displayed item to perform a function, that is different from the use of hovering the pointing device over the same displayed item to map its function to a received hotkey, is not disclosed or suggested by Slaunwhite (page 20, last paragraph).

Examiner disagrees.

As stated above, Slaunwhite teaches at least two ways of selecting the particular function for mapping a shortcut: a) typically by selecting item type instance a list on a customization dialog and b) when an item type instance is selected from a toolbar. Appellant, by failing to argue otherwise, seemly acknowledges that clicking and hovering are indeed obvious variants of each other when trying to indicate an action. Further, Appellant, seems to try and make the case the in Appellant's application the clicking performs one action while the hovering performs a different action, and that this would not have been known to one of ordinary skill in the art. This is wrong because: 1) Slaunwhite teaches that clicking over the same displayed item, as stated above, can be used to perform two different actions, for example a) (page 2, paragraph [0030]) performing a task and b) (page 2, paragraph [0037]) select a type instance from a list on a customization dialog), and thus clicking and hovering, being obvious variants of each

other, could clearly have been used to perform the two different actions; 2) one of ordinary skill in the art would have been well aware that clicking on an toolbar item might perform a task associated with that toolbar item, while hovering over that same toolbar item might perform the second action of providing a user with information regarding that toolbar item. Thus, and as stated in the rejection of these claims, one of ordinary skill in the art could have combined the elements as claimed by known methods (e.g., as taught by Slaunwhite and Forest), and in combination, each element merely would have performed the same function as it did separately (see KSR, 550 U.S. at ___, 82 USPQ2d at 1391).

12) The Examiner stated that Slaunwhite fails to specifically show the features of claim 10, but that Forest shows that it was well-known prior to the time of the invention to indicate one has reached a target by hovering, or not clicking, and selecting by clicking. However, Forest teaches selecting a displayed item (i.e. a key' s image) by dwelling or hovering. Such selection is the same as or redundant to selection by clicking, i.e., hovering is taught as simply a different way to select items, having the same function and effect as clicking. For example, Forest illustrates this equivalency by having a user select a key image by using a switch (a click) or by dwelling. It is not known or obvious to use selection by clicking for standard selection to perform a function, and selection by hovering over that same displayed item for a different function, i.e., to indicate the item' s function for mapping to a received hot key, where that indication does not include clicking the pointing device on the displayed item used for performing the function, as recited in claim 10 (page 21, first paragraph).

Examiner disagrees.

As stated above, Slaunwhite (page 2, paragraph [0030]) teaches clicking for both standard selection to perform a function, and (page 2, paragraph [0036]) for the different function of indicating the item' s function for mapping to a received hot key. As Appellant acknowledges, Forest teaches selecting a displayed item by dwelling or hovering and such selection being the same as or redundant to selection by clicking, which is similar to stating that clicking and hovering are obvious variations of each other. As one of ordinary skill in the art would readily understand, if clicking can be used to indicate two different functions (i.e., performing a task, and indicating a function for shortcut) for the same displayed item, then

clicking and hovering, being obvious variants of each other, could also be used to indicate the two different functions. Further, as one of ordinary skill in the art would readily recognize, clicking on a toolbar item might perform a first task associated with that toolbar item, while hovering over that same toolbar item might perform the second task of providing a user with information regarding that toolbar item. Thus, it was known to use selection by clicking for standard selection to perform a function, and selection by hovering over that same displayed item for a different function.

**(11) Related Proceeding(s) Appendix**

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Jordany Núñez/

Examiner, Art Unit 2175
11/3/2008

Conferees:

/Kieu D Vu/
Primary Examiner, Art Unit 2175

William Bashore

/WILLIAM L. BASHORE/
Supervisory Patent Examiner, Art Unit 2175

Dennis Chow

/DENNIS-DOON CHOW/
Supervisory Patent Examiner, Art Unit 2173